
Introducción a Ruby on Rails

— Moviendo la lógica de la vista —

Repaso

Crear una acción llamada: remember nuestro proyecto base.

Que imprima:

Hello, sample_name, of course we remember you!

Overview

- Cómo movemos la lógica de negocios de las vistas (views) al controlador(controller) con el objetivo de cumplir con el patrón MVC.
- Cuánto tiempo las variables de instancia de los controllers existen.

Actions methods (dentro del controller)

- Si el action no tiene código en el controller no tiene nada que hacer ahí - **podemos eliminarlo**
- Siempre que exista un **route definido apropiadamente**, y un **template correctamente nombrado**, el método del action no tiene nada que hacer ahí, ya que Rails puede encontrar el template correcto por convención.

Controller -> Nueva mirada

```
class GreeterController < ApplicationController  
  
  #def hello  
  #end  
  
  #def goodbye  
  #end  
  
end
```

< > ↺ localhost:3000/greeter/goodbye

Goodbye, iván

La hora es: 2016-10-30 23:20:27 -0300

Moviendo la lógica de negocios fuera de los Views

- Existe una razón por la cual nos es permitido especificar los actions en los controllers.
- Nuestra aplicación funciona, pero la lógica de negocios reside totalmente en la vista, y no está bien.
- Los views deben tener la menor cantidad posible de código ruby.

Moviendo la lógica de negocios fuera de los Views

- Las variables de instancia de los actions en los controllers están **disponibles en las vistas**. (Está dentro del scope).

```
class GreeterController < ApplicationController

  def hello
    random_names = ["vanessa", "iván", "eduardo"]
    @name = random_names.sample
    @time = Time.now
    @times_displayed ||= 0
    @times_displayed += 1
  end

  #def goodbye
  #end

end
```

Scope de una variable de instancia

```
<h1>Greetings, <%= @name %></h1>
<p>La hora es: <%= @time %></p>
<p>Esta página se ha mostrado en:
  <%= @times_displayed %>
  <%= @times_displayed == 1 ? "vez" : "veces"%>
</p>
```

< > ↺ localhost:3000/greeter/hello

Greetings, vanessa

La hora es: 2016-10-30 23:33:10 -0300

Esta página se ha mostrado en: 1 vez

1 las dos
veces?

< > ↺ localhost:3000/greeter/hello

Greetings, eduardo

La hora es: 2016-10-30 23:33:48 -0300

Esta página se ha mostrado en: 1 vez

Variables de instancia en Rails

- A diferencia de otros frameworks (los servlets por ejemplo), no se pueden **almacenar valores en las variables de instancia de los controllers entre los requests.**
- Cada nuevo request, creará nuevas variables de instancia.
- Es por eso que `@times_displayed` no funciona en nuestro ejemplo.
- Cuales son las alternativas?
 - Session
 - Database

Hablaremos de las variables de session y la base de datos más adelante!

Entonces...

- Mantengamos la lógica de negocios **fuera de las vistas** (views)
- **Las variables de instancia** de los controladores están **disponibles en la vista** (view)
- Las variables de instancia **solo viven lo que dura un request.**